# TROPICAL RAINFALL MEASURING MISSION PRECIPITATION PROCESSING SYSTEM

## File Specification
## 1B01

**Version 7**

March 22, 2012

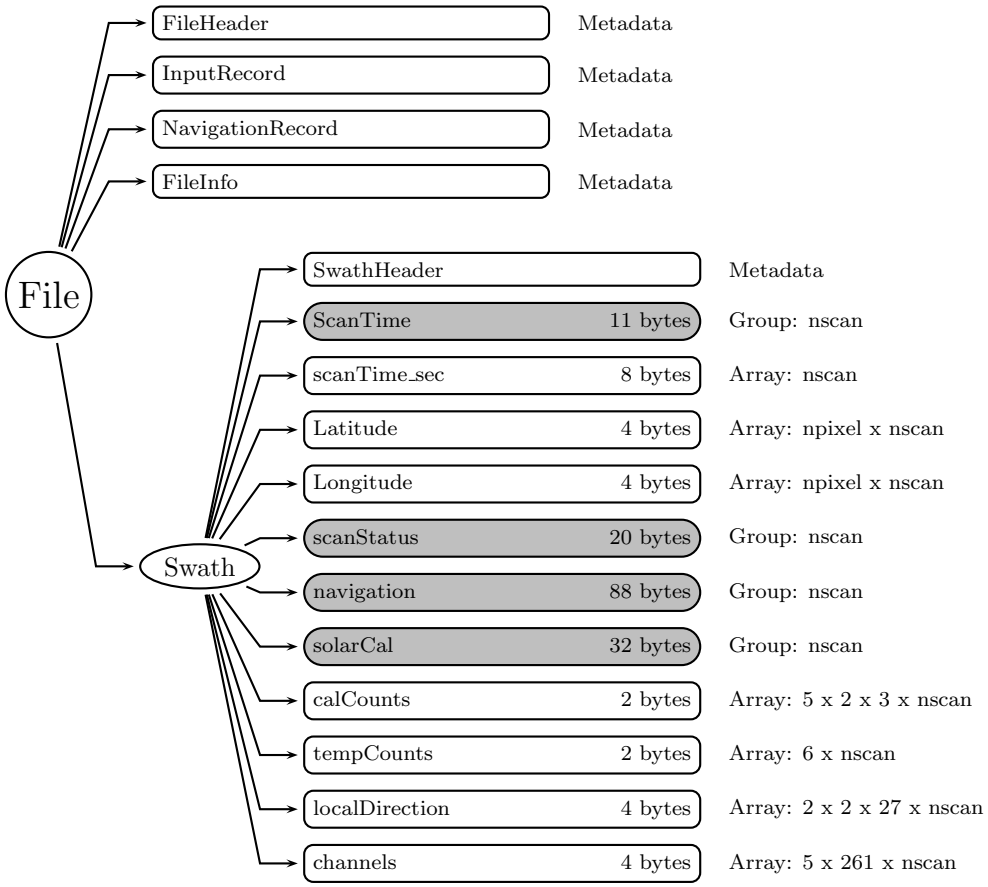


Figure 1: Data Format Structure for 1B01, VIRS Radiance

### 0.1 1B01 - VIRS Radiance

The VIRS Level-1B Product, 1B01, "VIRS Radiance," is written in HDF. The following sizing parameter is used in describing these formats:

Dimension definitions:

| | | |
|---|---|---|
| nscan | var | Number of scans in the granule. |
| npixel | 261 | Number of pixels in each scan. |

Figure 1 through Figure 5 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the

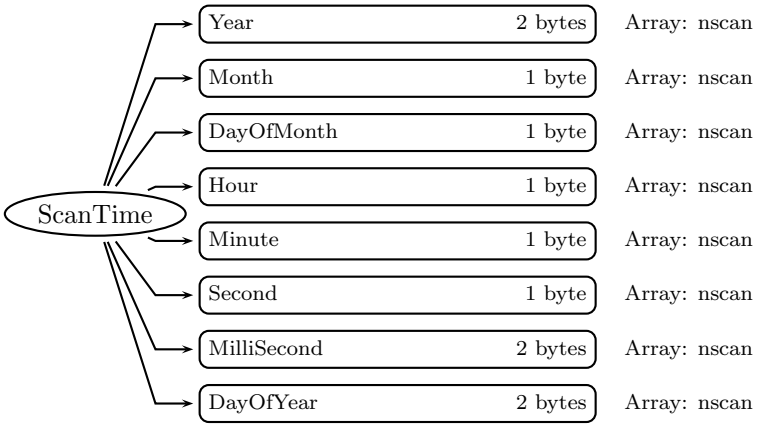


Figure 2: Data Format Structure for 1B01, ScanTime

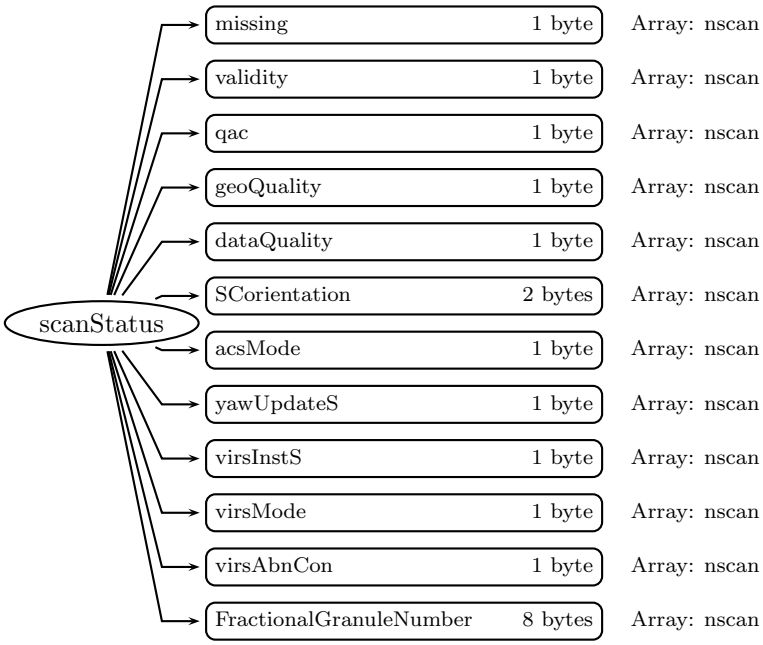


Figure 3: Data Format Structure for 1B01, scanStatus

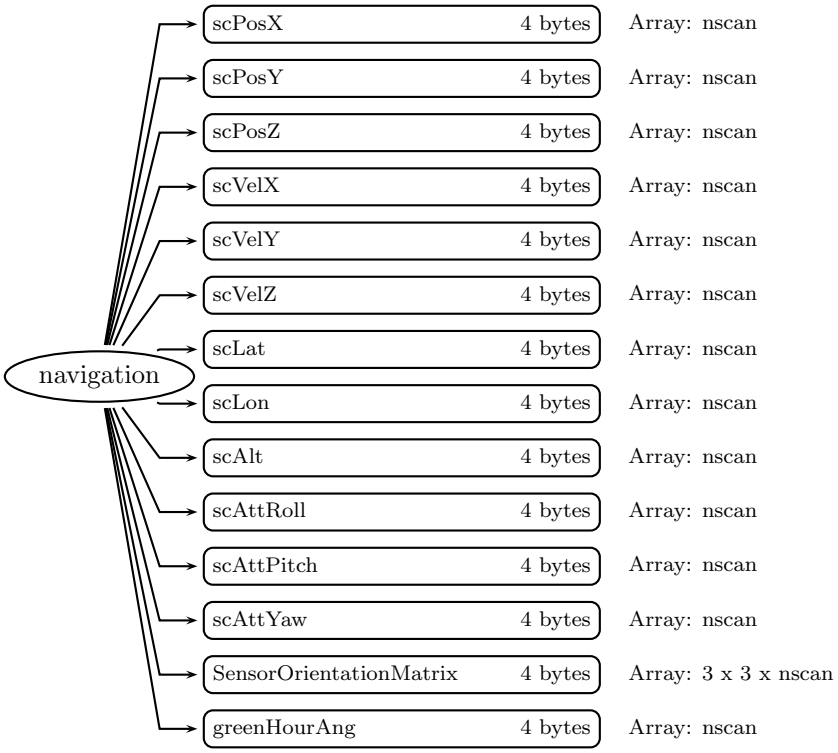


Figure 4: Data Format Structure for 1B01, navigation

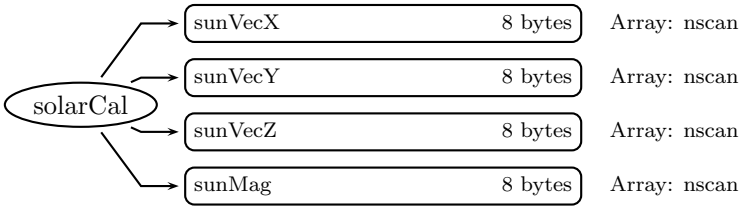


Figure 5: Data Format Structure for 1B01, solarCal

same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

## **Swath** (Swath)

**SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

## **ScanTime** (Group)

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**scanTime_sec** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

## **scanStatus** (Group)

**missing** (1-byte integer, array size: nscan):
Missing indicates whether information is contained in the scan data. The values are:

```
0 Scan data elements contain information
1 Scan was missing in the telemetry data
2 Scan data contains no elements with rain
```

**validity** (1-byte integer, array size: nscan):
Validity is a summary of status modes. If all status modes are routine, all bits in Validity = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. Validity does not assess data or geolocation quality. Validity is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

```
Bit Meaning if bit = 1
0   Spare (always 0)
1   Non-routine spacecraft orientation (2 or 3)
2   Non-routine ACS mode (other than 4)
3   Non-routine yaw update status (0 or 1)
4   Non-routine instrument status (Bit 0 = 0 or bit 1 = 0)
5   Non-routine QAC (non-zero)
6   Spare (always 0)
7   Spare (always 0)
```

**qac** (1-byte integer, array size: nscan):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

**geoQuality** (1-byte integer, array size: nscan):
geoQuality is broken into 8 one-bit flags. Some flags represent problems but other flags are informational. Bits 0, 5, and 6 represent problems: 0 = 'good' quality and 1 = 'bad' quality. It is recommended not to use scans when any problem flag is 1. The informational flags have: 0 = routine conditions and 1 = non-routine conditions. Bit 0 is the most significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**(7-i). Note that good scans may have non-zero geoQuality. Each flag is listed below.

```
Bit Meaning if bit = 1
0   Grossly bad geolocation results:
    Spacecraft position vector magnitude outside range 6715 to 6790 km.
    Z component of midpoint of scan outside range -4100 to 4100 km.
    Distance from S/C to midpoint of scan outside range 500 to 750 km.
1   Unexpectedly large scan to scan jumps in geolocated positions in along and
    cross track directions for first, middle, and last pixels in each scan.
    Allowed deviation from nominal jump in along track motion = 3.0 km (first
    pixel), 3.0 km (middle pixel), and 3.0 km (last pixel).
    Allowed deviation from nominal jump in cross track motion = 3.0 km (first
    pixel), 3.0 km (middle pixel), and 3.0 km (last pixel).
    Bit set in normal mode only.
2   Scan to scan jumps in yaw, pitch, and roll exceed maximum values.  Values
    are :  yaw = 0.005 radians; pitch = 0.005 radians; roll = 0.005 radians.
    Bit set in normal control mode only.
3   In normal mode, yaw outside range (-0.005, 0.005) radians; pitch outside
    range (-0.005, 0.005) radians; roll outside range (-0.005, 0.005) radians.
4   Satellite undergoing maneuvers during which geolocation will be less
    accurate.
5   Summary QA flag for dataQuality:  Set to 1 if bit 0 is 1 or bit 6 is 1,
    i.e. Grossly bad or failed geolocation calculations.
    Science data use not recommended.
6   Geolocation calculations failed (fill values inserted in the per pixel
    geolocation products, but not in metadata).
7   Missing attitude data.  ACS data gap larger than 1.0 seconds.
      Pitch, roll, and yaw are interpolated or extrapolated from nearby data.
```

**dataQuality** (1-byte integer, array size: nscan):
dataQuality is a flag for overall scan quality. Unless this is 0, the scan data is meaningless to higher science processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

```
Bit  Meaning if bit = 1
0    missing
5    geoQuality indicates bad or missing values
6    validity is not normal
```

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the TMI scan. If +X is forward, SCorientation is 0. If -X is forward, SCorientation is 180. If -Y is forward, SCorientation is 90. Values range from 0 to 360 degrees. Special values are defined as:
-8003 Inertial
-8004 Unknown
-9999 Missing value

**acsMode** (1-byte integer, array size: nscan):

```
Value Meaning
0     Standby
1     Sun Acquire
2     Earth Acquire
3     Yaw Acquire
4     Nominal
5     Yaw Maneuver
6     Delta-H (Thruster)
7     Delta-V (Thruster)
8     CERES Calibration
```

**yawUpdateS** (1-byte integer, array size: nscan):

```
Value Meaning
0     Inaccurate
1     Indeterminate
2     Accurate
```

**virsInstS** (1-byte integer, array size: nscan):

```
Value Meaning
0     Day (no calibration occurring)
1     Night
2     Monitor Scan Stability
3     Day with Calibration
```

**virsMode** (1-byte integer, array size: nscan):

```
Value Meaning
0     mission mode
1     safehold mode
2     outgas mode
3     activation mode
```

**virsAbnCon** (1-byte integer, array size: nscan):
Bit 0 is the most significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**(8-i) - 1).

```
Bit Value Meaning
0   0     normal
    1     scan phase error
1   0     normal
    1     selftest error
2   0     normal
    1     thermal data missing
3   0     normal
    1     moon in space view
4   0     normal
    1     H/K data drop-out suspected
5   0     normal
    1     SV counts for channel 4 or 5 greater than
          L1B01_MIN_DNSV
6   0     not used
7   0     not used
```

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

## **navigation** (Group)

**scPosX** (4-byte float, array size: nscan):
The x component of the position (m) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Geocentric Inertial Coordinates are also commonly known as Earth Centered Inertial coordinates. These coordinates will be True of Date (rather than Epoch 2000 which

are also commonly used), as interpolated from the data in the Flight Dynamics Facility ephemeris files generated for TRMM.

**scPosY** (4-byte float, array size: nscan):
The y component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

**scPosZ** (4-byte float, array size: nscan):
The z component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

**scVelX** (4-byte float, array size: nscan):
The x component of the velocity ($ms^{-1}$) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

**scVelY** (4-byte float, array size: nscan):
The y component of the velocity ($ms^{-1}$) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

**scVelZ** (4-byte float, array size: nscan):
The z component of the velocity ($ms^{-1}$) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

**scLat** (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time.

**scLon** (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time.

**scAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time.

**scAttRoll** (4-byte float, array size: nscan):
The satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

**scAttPitch** (4-byte float, array size: nscan):
The satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the

rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

**scAttYaw** (4-byte float, array size: nscan):
The satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

**SensorOrientationMatrix** (4-byte float, array size: 3 x 3 x nscan):
SensorOrientationMatrix is the rotation matrix from the instrument coordinate frame to Geocentric Inertial Coordinates at the Scan mid-Time. It is unitless.

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates.

## **solarCal** (Group)

**sunVecX** (8-byte float, array size: nscan):
Solar Position (X-component) (Geocentric Inertial Coord).

**sunVecY** (8-byte float, array size: nscan):
Solar Position (Y-component) (Geocentric Inertial Coord).

**sunVecZ** (8-byte float, array size: nscan):
Solar Position (Z-component) (Geocentric Inertial Coord).

**sunMag** (8-byte float, array size: nscan):
Sun-Earth Distance (m).

**calCounts** (2-byte integer, array size: 5 x 2 x 3 x nscan):
Raw calibration counts are given in four dimensions. The first dimension is the channel

number, the second dimension is the data word, the third dimension is blackbody, space view and solar diffuser, in that order, and the fourth dimension is the number of scans.

**tempCounts** (2-byte integer, array size: 6 x nscan):
Temperatures of the black body, primary and redundant, the radiant cooler temperatures, primary and redundant, the mirror temperature, and the electronics module temperature. All quantities have units of counts, and have minimum values of 0, and maximum values of 4095.

**localDirection** (4-byte float, array size: 2 x 2 x 27 x nscan):
Angles (degrees) to the satellite and sun from the IFOV pixel position on the earth are given in 4 dimensions. The first dimension is zenith and azimuth angles, in that order. The zenith angle is measured between the local pixel geodetic zenith and the direction to the satellite. The azimuth angle is measured clockwise from the local North direction around toward the local East direction. The second dimension is the object to which the directions point, namely the satellite and the sun, in that order. The third dimension is the pixel number. Angles are given only for every tenth pixel along a scan: pixel 1, 11, 21, ..., and 261. For the pixel dimension, Offset = 0 and Increment = -10. The fourth dimension is the scan number.

**channels** (4-byte float, array size: 5 x 261 x nscan):
Scene data for the channels, measured in Radiance ($mWcm^{-2}\mu m^{-1}sr^{-1}$). sr means steradian. The three dimensions are channel, pixel, and scan. The range, accuracy and wavelength for each channel are as follows:

| Channel | Minimum | Maximum | Accuracy | Wavelength (micrometers) |
|---|---|---|---|---|
| 1 | 0 | 65.5 | 10% | 0.63 |
| 2 | 0 | 32.7 | 10% | 1.6 |
| 3 | 0 | 0.111 | 2% | 3.75 |
| 4 | 0 | 1.371 | 2% | 10.8 |
| 5 | 0 | 1.15 | 2% | 12.0 |

## C Structure Header file:

```
#ifndef _TK_1B01_H_
#define _TK_1B01_H_

#ifndef _L1B01_SOLARCAL_
#define _L1B01_SOLARCAL_

typedef struct {
    double sunVecX;
    double sunVecY;
    double sunVecZ;
    double sunMag;
```

```
} L1B01_SOLARCAL;

#endif

#ifndef _L1B01_NAVIGATION_
#define _L1B01_NAVIGATION_

typedef struct {
    float scPosX;
    float scPosY;
    float scPosZ;
    float scVelX;
    float scVelY;
    float scVelZ;
    float scLat;
    float scLon;
    float scAlt;
    float scAttRoll;
    float scAttPitch;
    float scAttYaw;
    float SensorOrientationMatrix[3][3];
    float greenHourAng;
} L1B01_NAVIGATION;

#endif

#ifndef _L1B01_SCANSTATUS_
#define _L1B01_SCANSTATUS_

typedef struct {
    signed char missing;
    signed char validity;
    signed char qac;
    signed char geoQuality;
    signed char dataQuality;
    short SCorientation;
    signed char acsMode;
    signed char yawUpdateS;
    signed char virsInstS;
    signed char virsMode;
    signed char virsAbnCon;
    double FractionalGranuleNumber;
} L1B01_SCANSTATUS;
```

```
#endif

#ifndef _L1B01_SCANTIME_
#define _L1B01_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L1B01_SCANTIME;

#endif

#ifndef _L1B01_SWATH_
#define _L1B01_SWATH_

typedef struct {
    L1B01_SCANTIME ScanTime;
    double scanTime_sec;
    float Latitude[261];
    float Longitude[261];
    L1B01_SCANSTATUS scanStatus;
    L1B01_NAVIGATION navigation;
    L1B01_SOLARCAL solarCal;
    short calCounts[3][2][5];
    short tempCounts[6];
    float localDirection[27][2][2];
    float channels[261][5];
} L1B01_SWATH;

#endif

#endif
```

## Fortran Structure Header file:

```
STRUCTURE /L1B01_SOLARCAL/
```

```
    REAL*8 sunVecX
    REAL*8 sunVecY
    REAL*8 sunVecZ
    REAL*8 sunMag
END STRUCTURE

STRUCTURE /L1B01_NAVIGATION/
    REAL*4 scPosX
    REAL*4 scPosY
    REAL*4 scPosZ
    REAL*4 scVelX
    REAL*4 scVelY
    REAL*4 scVelZ
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 scAttRoll
    REAL*4 scAttPitch
    REAL*4 scAttYaw
    REAL*4 SensorOrientationMatrix(3,3)
    REAL*4 greenHourAng
END STRUCTURE

STRUCTURE /L1B01_SCANSTATUS/
    BYTE missing
    BYTE validity
    BYTE qac
    BYTE geoQuality
    BYTE dataQuality
    INTEGER*2 SCorientation
    BYTE acsMode
    BYTE yawUpdateS
    BYTE virsInstS
    BYTE virsMode
    BYTE virsAbnCon
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1B01_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
```

```
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L1B01_SWATH/
    RECORD /L1B01_SCANTIME/ ScanTime
    REAL*8 scanTime_sec
    REAL*4 Latitude(261)
    REAL*4 Longitude(261)
    RECORD /L1B01_SCANSTATUS/ scanStatus
    RECORD /L1B01_NAVIGATION/ navigation
    RECORD /L1B01_SOLARCAL/ solarCal
    INTEGER*2 calCounts(5,2,3)
    INTEGER*2 tempCounts(6)
    REAL*4 localDirection(2,2,27)
    REAL*4 channels(5,261)
END STRUCTURE
```